For written notes on this lecture, please read chapter 3 of *The Practical Bioinformatician,*

CS2220: Introduction to Computational Biology

Lecture 1: Essence of Knowledge Discovery

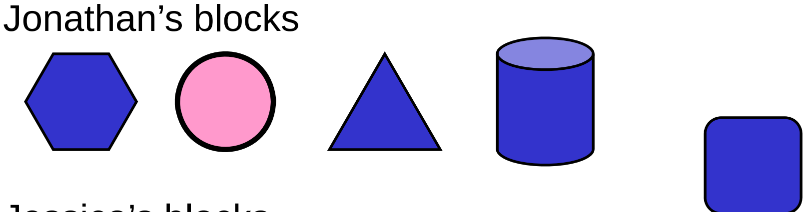
Lisa Tucker-Kellogg

14 January 2010

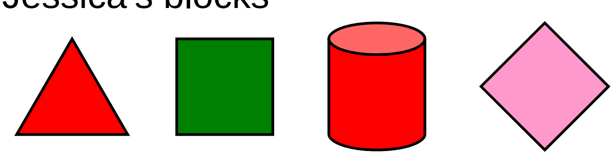
Slides thanks to Limsoon Wong 2009

What is Data Mining?

Jonathan's blocks



Jessica's blocks



Whose block is this?

Jonathan's rules	: Blue or Circle
Jessica's rules	: All the rest

What is Data Mining?

Question: Can you explain how?

The Steps of Data Mining

- **Training data gathering**
- **Feature generation**
  - k-grams, colour, texture, domain know-how, ...
- **Feature selection**
  - Entropy, $\chi 2$, CFS, t-test, domain know-how...
- **Feature integration**
  - SVM, ANN, PCL, CART, C4.5, kNN, ...

Some classifier/ methods

What is Accuracy?

What is Accuracy?

	predicted as positive	predicted as negative
positive	TP	FN
negative	FP	TN

$$\text{Accuracy} = \frac{\text{No. of correct predictions}}{\text{No. of predictions}}$$

$$= \frac{TP + TN}{TP + TN + FP + FN}$$

Examples (Balanced Population)

classifier	TP	TN	FP	FN	Accuracy
A	25	25	25	25	50%
B	50	25	25	0	75%
C	25	50	0	25	75%
D	37	37	13	13	74%

- **Clearly, B, C, D are all better than A**
- **Is B better than C, D?**
- **Is C better than B, D?**
- **Is D better than B, C?**

Accuracy may not tell the whole story

Examples (Unbalanced Population)

classifier	TP	TN	FP	FN	Accuracy
A	25	75	75	25	50%
B	0	150	0	50	75%
C	50	0	150	0	25%
D	30	100	50	20	65%

- **Clearly, D is better than A**
- **Is B better than A, C, D?**

Exercise: What is B's Prediction strategy?

High accuracy is meaningless if population is unbalanced

What is Sensitivity (aka Recall)?

	predicted as positive	predicted as negative
positive	TP	FN
negative	FP	TN

$$\underset{\text{wrt positives}}{\text{Sensitivity}} = \frac{\text{No. of correct positive predictions}}{\text{No. of positives}}$$

$$= \frac{TP}{TP + FN}$$

Sometimes sensitivity wrt negatives is termed **specificity**

Exercise: Write down the formula for specificity

What is Precision?

	predicted as positive	predicted as negative
positive	TP	FN
negative	FP	TN

$$\underset{\text{wrt positives}}{\text{Precision}} = \frac{\text{No. of correct positive predictions}}{\text{No. of positives predictions}}$$

$$= \frac{TP}{TP + FP}$$

Unbalanced Population Revisited

classifier	TP	TN	FP	FN	Accuracy	Sensitivity	Precision
A	25	75	75	25	50%	50%	25%
B	0	150	0	50	75%		
C	50	0	150	0	25%		
D	30	100	50	20	65%	60%	38%

- **What are the senstivity and precision of B and C?**
- **Is B better than A, C, D?**

Abstract Model of a Classifier

- **Given a test sample *S***
- **Compute scores *p(S), n(S)***
- **Predict *S* as negative if *p(S)* < *t* * *n(s)***
- **Predict *S* as positive if *p(S)* ≥ *t* * *n(s)***

t is the decision threshold of the classifier

changing *t* affects the recall and precision, and hence accuracy, of the classifier

An Example

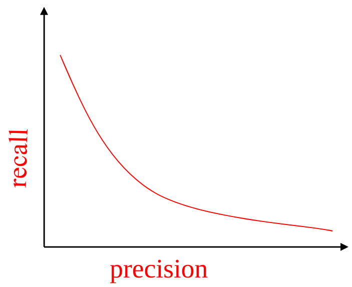
S	P(S)	N(S)	Actual Class	Predicted Class @ t = 3	Predicted Class @ t = 2
2	0.961252	0.038748	P	P	P
3	0.435302	0.564698	N	N	N
6	0.691596	0.308404	P	N	P
7	0.180885	0.819115	N	N	N
8	0.814909	0.185091	P	P	P
10	0.887220	0.112780	P	P	P
			accuracy	3/6	6/6
			recall	3/4	4/4
			precision	3/3	4/4

Recall that …

- **Predict *S* as negative if $p(S) < t * n(s)$**
- **Predict *S* as positive if $p(S) \geq t * n(s)$**

Precision-Recall Trade-off

- A predicts better than B if A has better recall and precision than B
- There is a trade-off between recall and precision



- In some applications, once you reach a satisfactory precision, you optimize for recall
- In some applications, once you reach a satisfactory recall, you optimize for precision

Exercise: Why is there a trade off betw recall and precision?

Comparing Prediction Performance

- **Accuracy is the obvious measure**
  - But it conveys the right intuition only when the positive and negative populations are roughly equal in size
- **Recall and precision together form a better measure**
  - But what do you do when A has better recall than B and B has better precision than A?

So let us look at some alternate measures ….

F-Measure (Used in Info Extraction)

- **Take the harmonic mean of recall and precision**

$$F = \frac{2 * \text{recall} * \text{precision}}{\text{recall} + \text{precision}} \quad \text{(wrt positives)}$$

classifier	TP	TN	FP	FN	Accuracy	F-measure
A	25	75	75	25	50%	33%
B	0	150	0	50	75%	undefined
C	50	0	150	0	25%	40%
D	30	100	50	20	65%	46%

Does not accord with intuition:
C predicts everything as +ve, but still rated better than A

Adjusted Accuracy

- **Weigh by the importance of the classes**

Adjusted accuracy = α * Sensitivity + β * Specificity

where $\alpha + \beta = 1$

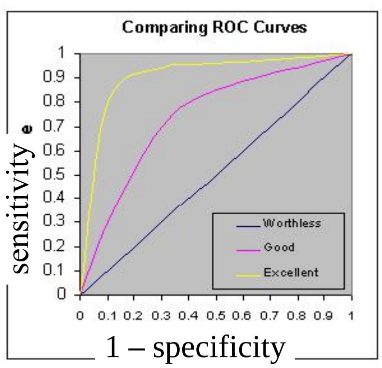
typically, $\alpha = \beta = 0.5$

classifier	TP	TN	FP	FN	Accuracy	Adj Accuracy
A	25	75	75	25	50%	50%
B	0	150	0	50	75%	50%
C	50	0	150	0	25%	50%
D	30	100	50	20	65%	63%

But people can't always agree on values for α, β

ROC Curves

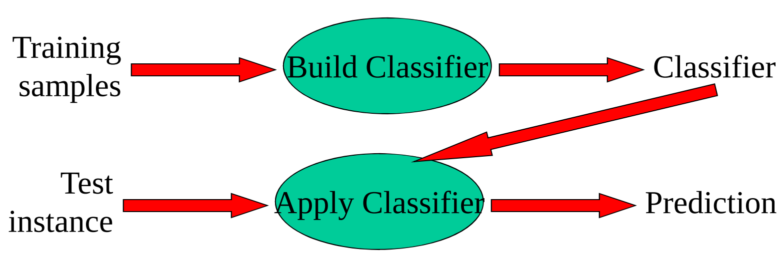
- **By changing t, we get a range of sensitivities and specificities of a classifier**
- **A predicts better than B if A has better sensitivities than B at most specificities**
- **Leads to ROC curve that plots sensitivity vs. (1 – specificity)**
- **Then the larger the area under the ROC curve, the better**



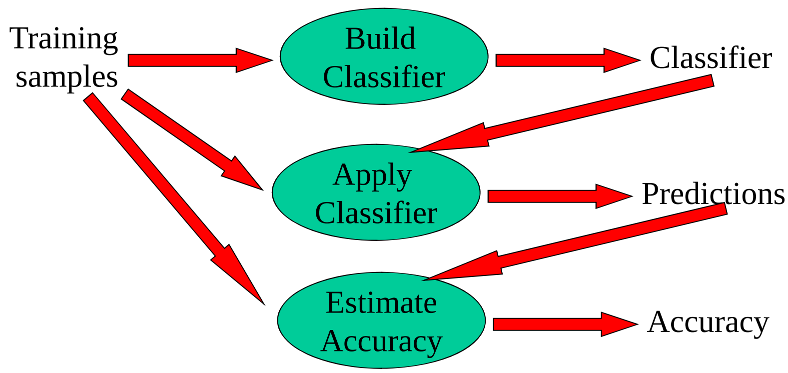
Exercise: Draw a typical curve of sensitivity vs specificity

What is Cross Validation?

Construction of a Classifier



Estimate Accuracy: Wrong Way



Exercise: Why is this way of estimating accuracy wrong?

Recall ...

…the abstract model of a classifier

- **Given a test sample S**
- **Compute scores p(S), n(S)**
- **Predict S as negative if p(S) < t * n(s)**
- **Predict S as positive if p(S) ≥ t * n(s)**

t is the decision threshold of the classifier

K-Nearest Neighbour Classifier (k-NN)

- **Given a sample S, find the k observations S_i in the known data that are “closest” to it, and average their responses**
- **Assume S is well approximated by its neighbours**

$$p(S) = \sum_{S_i \in N_k(S) \cap D^P} 1 \qquad n(S) = \sum_{S_i \in N_k(S) \cap D^N} 1$$

where $N_k(S)$ is the neighbourhood of S defined by the k nearest samples to it.

Assume distance between samples is Euclidean distance for now

Illustration of kNN (k=8)

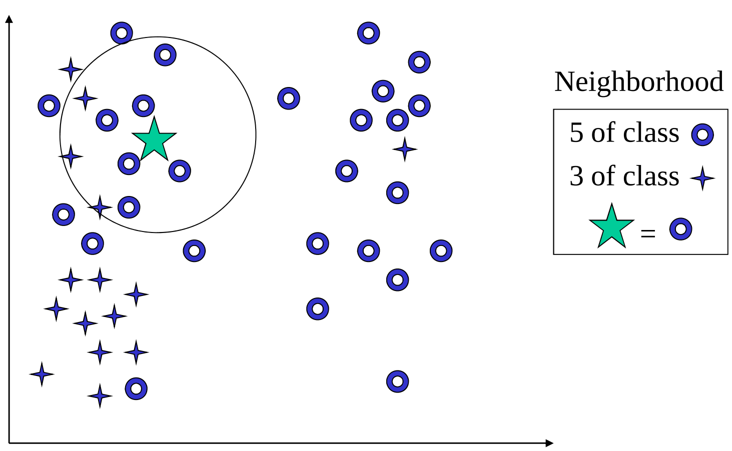
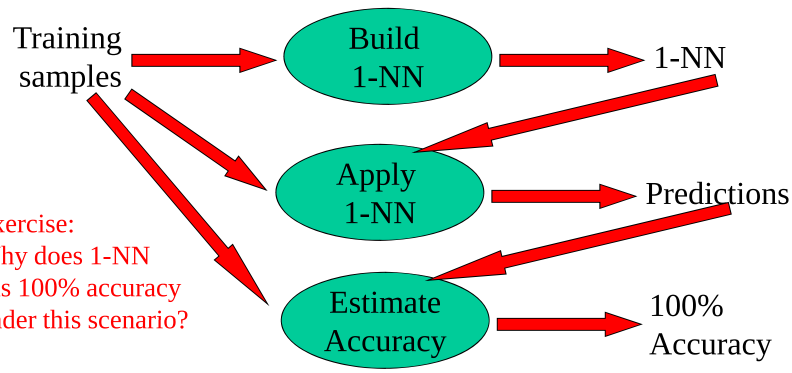


Image credit: Zaki

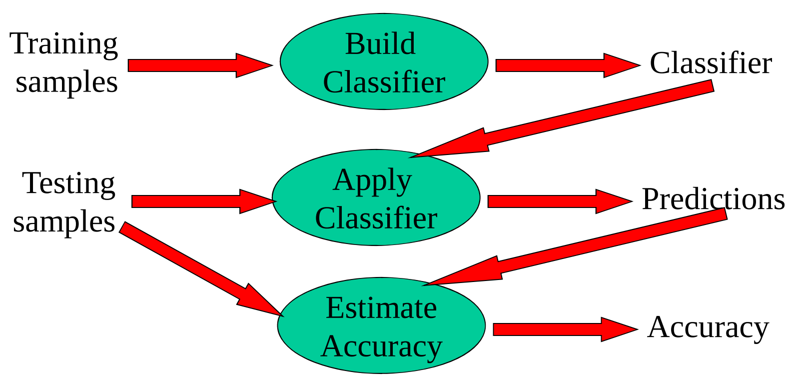
Estimate Accuracy: Wrong Way



Exercise:
Why does 1-NN
has 100% accuracy
under this scenario?

For sure k-NN (k = 1) has 100% accuracy in the "accuracy estimation" procedure above. But does this accuracy generalize to new test instances?

Estimate Accuracy: Right Way



Testing samples are NOT to be used during "Build Classifier"

How Many Training and Testing Samples?

- No fixed ratio between training and testing samples; but typically 2:1 ratio
- Proportion of instances of different classes in testing samples should be similar to proportion in training samples

- What if there are insufficient samples to reserve 1/3 for testing?
- Ans: Cross validation

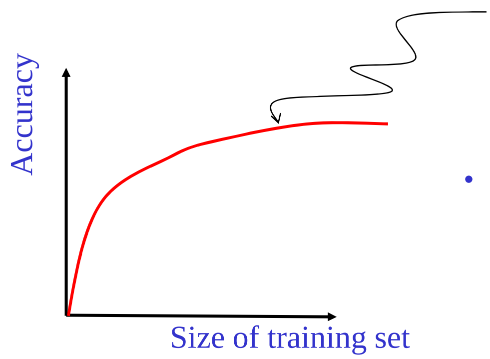
Cross Validation

1.Test	2.Train	3.Train	4.Train	5.Train
1.Train	2.Test	3.Train	4.Train	5.Train
1.Train	2.Train	3.Test	4.Train	5.Train
1.Train	2.Train	3.Train	4.Test	5.Train
1.Train	2.Train	3.Train	4.Train	5.Test

- Divide samples into k roughly equal parts
- Each part has similar proportion of samples from different classes
- Use each part to test other parts
- Total up accuracy

How Many Fold?

- **If samples are divided into k parts, we call this k-fold cross validation**



- **Choose k so that**
  - the k-fold cross validation accuracy does not change much from k-1 fold
  - each part within the k-fold cross validation has similar accuracy
- **k = 5 or 10 are popular choices for k**

Bias and Variance

Suppose a butcher weighs a steak with his thumb on the scale. That causes an error in the measurement, but little has been left to chance. Take another example. Suppose a drapery store uses a cloth tape measure which has stretched from 36 inches to 37 inches in length. Every "yard" of cloth they sell to a customer has an extra inch tacked onto it. This isn't a chance error, because it always works for the customer. The butcher's thumb and the stretched tape are two examples of *bias*, or *systematic error*.

> Bias affects all measurements the same way, pushing them in the same direction. Chance errors change from measurement to measurement, sometimes up and sometimes down.

The basic equation has to be modified when each measurement is thrown off by bias as well as chance error:

individual measurement = exact value + bias + chance error.

If there is no bias in a measurement procedure, the long-run average of repeated measurements should give the exact value of the thing being measured: the

Source: Freedman et al., *Statistics*, Norton, 1998

Bias-Variance Decomposition

- **Suppose classifiers C_j and C_k were trained on different sets S_j and S_k of 1000 samples each**
- **Then C_j and C_k might have different accuracy**
- **What is the expected accuracy of a classifier C trained this way?**
- **Let Y = f(X) be what C is trying to predict**
- **The expected squared error at a test instance x, averaging over all such training samples, is**

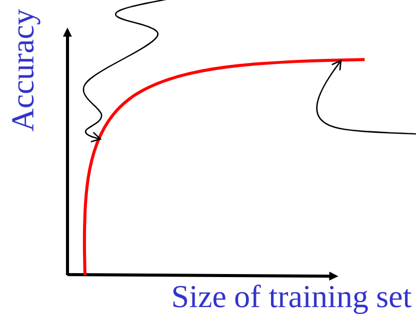
$$E[C(x) - f(x)]^2$$
$$= E[C(x) - E[C(x)]]^2$$
$$+ [E[C(x)] - f(x)]^2$$

Variance: how much our estimate C(x) will vary across the different training sets

Bias: how far is our ave prediction E[C(x)] from the truth

Bias-Variance Trade-Off

- **In k-fold cross validation,**
  - small k tends to under estimate accuracy (i.e., large bias downwards)
  - Large k has smaller bias, but can have high variance



Curse of Dimensionality

Recall ...

…the abstract model of a classifier

- **Given a test sample S**
- **Compute scores p(S), n(S)**
- **Predict S as negative if p(S) < t * n(s)**
- **Predict S as positive if p(S) ≥ t * n(s)**

t is the decision threshold of the classifier

K-Nearest Neighbour Classifier (k-NN)

- **Given a sample S, find the k observations S_i in the known data that are "closest" to it, and average their responses**
- **Assume S is well approximated by its neighbours**

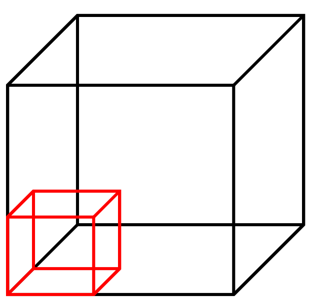
$$p(S) = \sum_{S_i \in N_k(S) \cap D^P} 1 \qquad n(S) = \sum_{S_i \in N_k(S) \cap D^N} 1$$

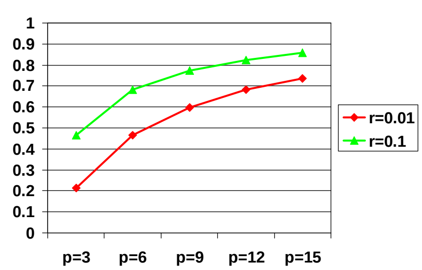
where $N_k(S)$ is the neighbourhood of S defined by the k nearest samples to it.

Assume distance between samples is Euclidean distance for now

Curse of Dimensionality

- **How much of each dimension is needed to cover a proportion r of total sample space?**
- **Calculate by $e_p(r) = r^{1/p}$**
- **So, to cover 10% of a 15-D space, need 85% of each dimension!**





Exercise: Why $\mathbf{e_p(r) = r^{1/p}}$?

Consequence of the Curse

- Suppose the number of samples given to us in the total sample space is fixed
- Let the dimension increase
- Then the distance of the k nearest neighbours of any point increases
- Then the k nearest neighbours are less and less useful for prediction, and can confuse the k-NN classifier

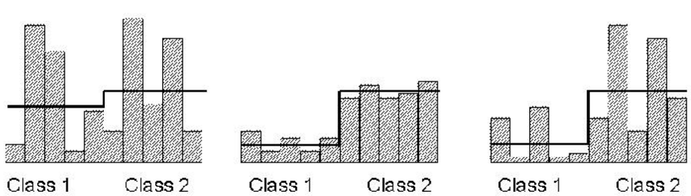
What is Feature Selection?

Tackling the Curse

- Given a sample space of p dimensions
- It is possible that some dimensions are irrelevant
- Need to find ways to separate those dimensions (aka features) that are relevant (aka signals) from those that are irrelevant (aka noise)

Signal Selection (Basic Idea)

- Choose a feature w/ low intra-class distance
- Choose a feature w/ high inter-class distance



Exercise: Name 2 well-known signal selection statistics

Signal Selection (e.g., t-statistics)

The t-stats of a signal is defined as

$$t = \frac{|\mu_1 - \mu_2|}{\sqrt{(\sigma_1^2/n_1) + (\sigma_2^2/n_2)}}$$

where σ_i^2 is the variance of that signal in class i, μ_i is the mean of that signal in class i, and n_i is the size of class i.

Self-fulfilling Oracle

- Construct artificial dataset with 100 samples, each with 100,000 randomly generated features and randomly assigned class labels
- Select 20 features with the best t-statistics (or other methods)
- Evaluate accuracy by cross validation using only the 20 selected features
- The resultant estimated accuracy can be ~90%
- But the true accuracy should be 50%, as the data were derived randomly

What Went Wrong?

- The 20 features were selected from the whole dataset
- Information in the held-out testing samples has thus been "leaked" to the training process
- The correct way is to re-select the 20 features at each fold; better still, use a totally new set of samples for testing

Concluding Remarks

What have we learned?

- **Methodology of data mining**
  - Feature generation, feature selection, feature integration
- **Evaluation of classifiers**
  - Accuracy, sensitivity, precision
  - Cross validation
- **Curse of dimensionality**
  - Feature selection concept
  - Self-fulfilling oracle

Any Questions?

Acknowledgements

- **The first two slides were shown to Prof. Wong 10+ years ago by Tan Ah Hwee**
- **These slides were assembled by Prof. Wong during years of teaching CS2220**

References

- John A. Swets, Measuring the accuracy of diagnostic systems, *Science* 240:1285--1293, June 1988
- Trevor Hastie et al., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2001. Chapters 1, 7
- Lance D. Miller et al., Optimal gene expression analysis by microarrays, *Cancer Cell* 2:353--361, 2002
- David Hand et al., *Principles of Data Mining*, MIT Press, 2001
- Jinyan Li et al., Data Mining Techniques for the Practical Bioinformatician, *The Practical Bioinformatician*, Chapter 3, pages 35—70, WSPC, 2004